04_modeling_opec_price_forecasting

December 2, 2025

1 Modeling OPEC Sentiment vs PP Prices

Predict next-month PP_EU using GPT comparison scores, hybrid index, FinBERT sentiment, and keyword densities.

[1]:

```
from pathlib import Path
BASE_DIR = Path.cwd()
if BASE_DIR.name == 'notebooks':
    BASE_DIR = BASE_DIR.parent

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics import mean_absolute_error
from sklearn.ensemble import RandomForestRegressor
from xgboost import XGBRegressor

sns.set_style('whitegrid')

data_path = BASE_DIR / 'data' / 'processed' / 'master_opec_price_model_dataset.
  ↪csv'
df = pd.read_csv(data_path)
df['date'] = pd.to_datetime(df['date'])
df = df.sort_values('date').reset_index(drop=True)
df.head()
```

[1]:

```
        date  comparison_score  hybrid_index  finbert_sentiment  supply_up  \
0 2019-01-31               0.0           0.0           0.110262   0.004327
1 2019-02-28              -0.7          -0.7           0.046981   0.002033
2 2019-03-31              -0.7          -1.4           0.061729   0.008493
3 2019-04-30              -0.6          -2.0          -0.239369   0.000517
4 2019-05-31              -0.3          -2.3          -0.102873   0.003960

   supply_down  demand_up  demand_down  price_up  price_down   PP_EU  Brent  \
0     0.004103   0.002233     0.004360  0.000000    0.001540  1385.0  59.77
1     0.004222   0.000220     0.004228  0.000097    0.001873  1385.0  63.63
2     0.003438   0.003655     0.004500  0.000073    0.001865  1410.0  66.66
```

```
3     0.001615   0.000050     0.001818  0.000805    0.001073  1430.0  71.03
4     0.004860   0.002240     0.004933  0.000025    0.000148  1430.0  70.93

     WTI  NatGas  PP_EU_next_month  Brent_next_month
0  51.07    3.15            1385.0             63.63
1  54.53    2.69            1410.0             66.66
2  57.62    2.83            1430.0             71.03
3  63.22    2.62            1430.0             70.93
4  61.76    2.60            1430.0             63.35
```

1.1 Dataset overview

Columns include raw GPT comparison scores, cumulative hybrid index, FinBERT sentiment, keyword densities, and market prices aligned to month-end.

[2]:

```
df.describe(include='all').T
```

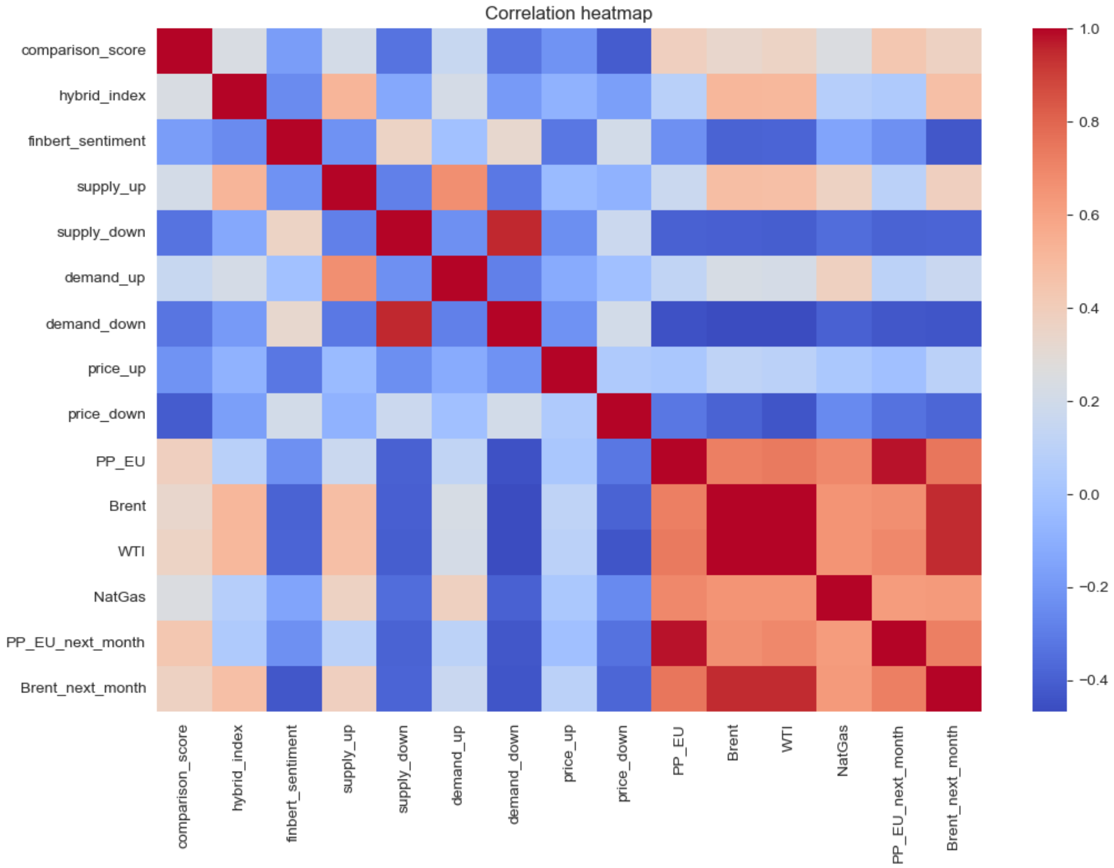
[2]:

```
                  count                 mean                  min  \
date                 80  2022-05-26 19:12:00  2019-01-31 00:00:00
comparison_score   80.0              0.11625                 -0.9
hybrid_index       80.0             -0.59625                -10.8
finbert_sentiment  80.0            -0.046661            -0.245568
supply_up          80.0              0.00598             0.000517
supply_down        80.0             0.002217               0.0002
demand_up          80.0             0.002424              0.00005
demand_down        80.0             0.002306             0.000265
price_up           80.0             0.000221                  0.0
price_down         80.0             0.000401                  0.0
PP_EU              80.0             1565.125                995.0
Brent              80.0            72.775875                27.29
WTI                80.0             68.38275                17.42
NatGas             80.0              3.35625                 1.71
PP_EU_next_month   80.0             1565.875                995.0
Brent_next_month   80.0            72.829875                27.29

                                   25%                  50%  \
date               2020-09-22 12:00:00  2022-05-15 12:00:00
comparison_score                 -0.45                 0.35
hybrid_index                    -5.525                -0.75
finbert_sentiment            -0.119719            -0.057719
supply_up                     0.003641             0.006189
supply_down                   0.001404             0.001944
demand_up                     0.001324             0.002092
demand_down                   0.001469             0.001999
price_up                      0.000025             0.000046
price_down                    0.000048             0.000112
PP_EU                           1385.0              1451.25
```

```
Brent                           63.855               73.555
WTI                            57.4775               70.375
NatGas                          2.3675                2.745
PP_EU_next_month                1385.0              1451.25
Brent_next_month                 64.05               73.555

                                   75%                  max         std
date               2024-02-07 06:00:00  2025-09-30 00:00:00         NaN
comparison_score                   0.6                  0.8    0.563094
hybrid_index                      5.05                  9.3    5.840007
finbert_sentiment             0.027162             0.300448    0.114086
supply_up                     0.008089             0.012422    0.002935
supply_down                   0.002726             0.008483    0.001372
demand_up                     0.003445              0.00717    0.001556
demand_down                   0.002699             0.007567     0.00139
price_up                      0.000456             0.000958    0.000294
price_down                     0.00064             0.001873    0.000494
PP_EU                           1722.5               2475.0  376.825703
Brent                            83.27               118.14   18.021506
WTI                             78.865               115.19   18.177425
NatGas                           3.735                 8.71     1.62051
PP_EU_next_month                1722.5               2475.0  376.522248
Brent_next_month                 83.27               118.14   17.988483
```

[3]:

```
plt.figure(figsize=(12, 8))
corr = df.corr(numeric_only=True)
sns.heatmap(corr, cmap='coolwarm', annot=False)
plt.title('Correlation heatmap')
plt.show()
```



1.2 Train/validation split

Use the earliest 80% of months for training and the latest 20% for validation to respect time order.

```
[4]: feature_cols = [
         'comparison_score', 'hybrid_index', 'finbert_sentiment',
         'supply_up', 'supply_down', 'demand_up', 'demand_down',
         'price_up', 'price_down', 'Brent', 'WTI', 'NatGas', 'PP_EU',
     ]

     target = 'PP_EU_next_month'
     split_idx = int(len(df) * 0.8)
     X_train, X_val = df.loc[:split_idx - 1, feature_cols], df.loc[split_idx:,␣
      ↪feature_cols]
     y_train, y_val = df.loc[:split_idx - 1, target], df.loc[split_idx:, target]
     dates_val = df.loc[y_val.index, 'date']

     naive_pred = df.loc[y_val.index, 'PP_EU']
     naive_mae = mean_absolute_error(y_val, naive_pred)
     print('Naive MAE (predict current PP as next month):', round(naive_mae, 2))
```

```
Naive MAE (predict current PP as next month): 11.25
```

[5]:

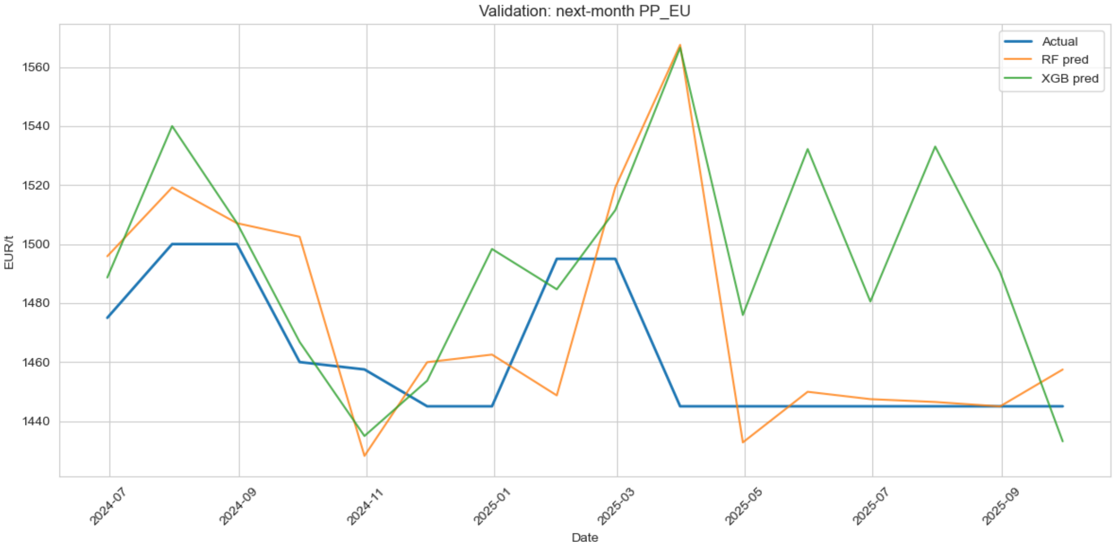
```python
rf = RandomForestRegressor(n_estimators=300, random_state=42, min_samples_leaf=2)
rf.fit(X_train, y_train)
rf_pred = rf.predict(X_val)
rf_mae = mean_absolute_error(y_val, rf_pred)

xgb = XGBRegressor(
    random_state=42, n_estimators=400, learning_rate=0.05, max_depth=4,
    subsample=0.9, colsample_bytree=0.9
)
xgb.fit(X_train, y_train)
xgb_pred = xgb.predict(X_val)
xgb_mae = mean_absolute_error(y_val, xgb_pred)

results = pd.DataFrame(
    [
        ['Naive (current PP)', naive_mae],
        ['RandomForest', rf_mae],
        ['XGBoost', xgb_mae],
    ],
    columns=['model', 'mae']
)
results
```

[5]:

```
                model        mae
0  Naive (current PP)  11.250000
1        RandomForest  23.632629
2             XGBoost  37.491753
```

[6]:

```python
val_df = pd.DataFrame({
    'date': dates_val,
    'actual': y_val.values,
    'rf_pred': rf_pred,
    'xgb_pred': xgb_pred,
})

plt.figure(figsize=(12, 6))
plt.plot(val_df['date'], val_df['actual'], label='Actual', linewidth=2)
plt.plot(val_df['date'], val_df['rf_pred'], label='RF pred', alpha=0.8)
plt.plot(val_df['date'], val_df['xgb_pred'], label='XGB pred', alpha=0.8)
plt.legend()
plt.title('Validation: next-month PP_EU')
plt.xlabel('Date')
plt.ylabel('EUR/t')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```


Validation: next-month PP_EU
Actual
RF pred
XGB pred
EUR/t
Date